CLAIMS:

1. A method of programming a design tool to implement a message processing system using an integrated circuit, the method comprising:

defining first attributes for a plurality of threads within said integrated circuit;

defining second attributes for a memory associated with said integrated circuit;

defining third attributes for an interconnection topology associated with at least one of said memory and said plurality of threads; and

defining fourth attributes for an interface to at least one of said memory and said plurality of threads.

2. The method of claim 1, wherein each of said first, second, third, and fourth attributes comprises at least one of functional attributes and architectural attributes.

3. The method of claim 1, wherein each of said first, second, third, and fourth attributes is defined using a set of primitives.

4. The method of claim 3, wherein said primitives comprise program code for driving said design tool.

5. The method of claim 4, wherein said program code is callable by a second design tool.

6. The method of claim 3, wherein said primitives comprise descriptions configured for interpretation by said design tool.

7. The method of claim 6, wherein said descriptions comprise extensible markup language (XML) descriptions.

8. The method of claim 1, wherein said step of defining said first attributes comprises:

specifying an instruction set for each of said plurality of threads.

9. The method of claim 8, wherein said instruction set is configured to define inter-thread communication amongst said plurality of threads.

10. The method of claim 8, wherein each of said plurality of threads comprises a state machine, and wherein said instruction set of each of said plurality of threads is associated with states of said state machine.

11. The method of claim 1, wherein said integrated circuit is a programmable logic device, and wherein said plurality of threads is implemented within programmable logic of said programmable logic device.

12. The method of claim 11, wherein at least one of said plurality of threads comprises a state machine circuit.

13. The method of claim 11, wherein at least one of said plurality of threads is implemented using a processor embedded within said programmable logic device.

14. The method of claim 1, wherein said step of defining said first attributes comprises:

including a function block to implement a thread of said plurality of threads.

15. The method of claim 1, wherein said step of defining said first attributes comprises:

defining an instruction set for a thread of said plurality of threads, said instruction set defining communication with an interface logic block.

16. The method of claim 1, further comprising at least one of:

defining fifth attributes for signal groups within said message processing system;

defining sixth attributes for run-time reconfiguration of said integrated circuit;

defining seventh attributes for implementation metrics associated with said message processing system; and

defining eighth attributes for debugging information for said message processing system.

17. A method of providing a programming interface for a design tool, said design tool adapted to implementing a message processing system using an integrated circuit, the method comprising:

providing a first set of primitives to specify attributes for a plurality of threads;

providing a second set of primitives to specify attributes for a memory associated with said integrated circuit;

providing a third set of primitives to specify an interconnection topology associated with at least one of said memory and said plurality of threads; and

providing a fourth set of primitives to specify attributes for an interface to at least one of said memory and said plurality of threads.

18. The method of claim 17, wherein said attributes specified by each of said first, second, third, and fourth

primitives comprises at least one of functional attributes and architectural attributes.

19. The method of claim 17, further comprising at least one of:

providing a fifth set of primitives to specify attributes for grouping signals within said message processing system;

providing a sixth set of primitives to specify attributes for run-time reconfiguration of said integrated circuit;

providing a seventh set of primitives to specify attributes for implementation metrics for said message processing system; and

providing an eighth set of primitives to specify attributes for debugging information associated with said message processing system.

20. The method of claim 17, wherein said first, second, third, and fourth sets of primitives comprise program code for driving said design tool.

21. The method of claim 20, wherein said program code is callable by a second design tool.

22. The method of claim 17, wherein said first, second, third, and fourth sets of primitives comprise descriptions configured for interpretation by said design tool.

23. The method of claim 22, wherein said descriptions comprise extensible markup language (XML) descriptions.

24. A computer readable medium having stored thereon an application programming interface for driving a design tool for designing a message processing system for implementation

using an integrated circuit, the application programming interface comprising:

a first set of primitives to specify attributes for a plurality of threads;

a second set of primitives to specify attributes for a memory associated with said integrated circuit;

a third set of primitives to specify an interconnection topology associated with at least one of said memory and said plurality of threads; and

a fourth set of primitives to specify attributes for an interface to at least one of said memory and said plurality of threads.